

BÁO CÁO ĐỒ ÁN 02: LOGIC BẬC NHẤT.

A. Thông tin nhóm và bảng phân công việc.

1. Thông tin nhóm.

MSSV	Họ và Tên
1712704	Nguyễn Phước Quang
1712302	Bùi Lý Chung
18120621	Nguyễn Đức Trục

2. Đánh giá công việc.

MSSV	Công việc	Mức độ hoàn thành.
1712704	Làm quen với công cụ prolog Tìm hiểu môi trường SWI-Prolog Xây dựng cây tri thức quản lý thư viện – Câu 1	100 %
1712302	Câu 2	100%
18120621	Cài đặt hệ thống suy diễn logic bằng ngôn ngữ lập trình	70%

B. Nội dung.

1. Làm quen với công cụ Prolog.

1.1 Prolog là ngôn ngữ lập trình logic.

Prolog là ngôn ngữ được sử dụng phổ biến nhất trong dòng các ngôn ngữ lập trình logic (Prolog có nghĩa là Programming in Logic). Ngôn ngữ Prolog do giáo sư người Pháp Alain Colmerauer và nhóm nghiên cứu của ông đề xuất lần đầu tiên tại trường Đại học Marseille đầu những năm 1970. Đến năm 1980, Prolog nhanh chóng được áp dụng rộng rãi ở châu Âu, được người Nhật chọn làm ngôn ngữ phát triển dòng máy tính thế hệ 5. Prolog đã được cài đặt trên các máy vi tính Apple II, IBM-PC, Macintosh.

Prolog còn được gọi là ngôn ngữ lập trình ký hiệu (symbolic programming) tương tự các ngôn ngữ lập trình hàm (functional programming), hay lập trình phi số (nonnumerical programming). Prolog rất thích hợp để giải quyết các bài toán liên quan đến các đối tượng (object) và mối quan hệ (relation) giữa chúng.

Prolog được sử dụng phổ biến trong lĩnh vực trí tuệ nhân tạo. Nguyên lý lập trình logic dựa trên các mệnh đề Horn (Horn logic). Một mệnh đề Horn biểu diễn một sự kiện hay một sự việc nào đó là đúng hoặc không đúng, xảy ra hoặc không xảy ra (có hoặc không có, v.v...).S

Ví dụ 1.1 sau đây là các mệnh đề Horn.

Nếu trời có mây đen thì trời sắp mưa.(1)

Petter là anh của Tom.(2)

Socrat là người.(3)

Nếu Nhi có tiền thì Nhi mua xe máy.(4)

Nếu Trinh không làm bài tập thì Trinh rớt môn.(5)

Trong mệnh đề horn trên , có mệnh đề (1) , (4) và (5) được gọi là các luật (rule), các mệnh đề còn lại được gọi là các sự kiện (fact). Một chương trình logic có thể xem như là một cơ sở dữ liệu gồm các mệnh đề Horn, hoặc dạng luật, hoặc dạng sự kiện và luật. Người sử dụng gọi chương trình logic bằng cách đặt câu hỏi truy vấn trên cơ sở dữ liệu.

1.2 Các kiểu dữ liệu trong Prolog.

Clause<mệnh đề>: các mệnh đề cấu trúc tạo nên chương trình Prolog.

Predicate<vị từ>: Mỗi mệnh đề được xây dựng từ các vị từ. Một vị từ là một phát biểu về các đối tượng đúng (True) hay là sai (False).

Symbols.

	Predicate Calculus	Prolog
and	\wedge	,
or	\vee	;
if	$-->$	$:-$
not	\sim	not

Variables and Names.

- Các biến bắt đầu bằng chữ hoa.
- Tên vị ngữ, tên chức năng và tên cho các đối tượng phải bắt đầu bằng một chữ cái thường. Quy tắc cấu tạo tên cũng giống như đối với phép tính vị ngữ.

Facts.

Thực tế là một biểu thức vị từ đưa ra một tuyên bố khai báo về miền vấn đề. Bất cứ khi nào một biến xuất hiện trong biểu thức Prolog, nó được giả định là được định lượng chung. Lưu ý rằng tất cả các câu Prolog phải kết thúc bằng dấu chấm.

Ví dụ:

likes(john, susie). /* John thích Susie */

likes(X, susie).	/* mọi người thích Susie */
likes(john, Y).	/* John thích mọi người */

Mệnh đề Horn: <Head>:-<Body>. Dùng để xây dựng các vị từ và được kết thúc bằng dấu chấm.

Head:-Body.

Ví dụ: ThiRot:- KhongHocBai.

Rule. Quy tắc là một biểu thức vị từ sử dụng hàm ý logic (:-) để mô tả mối quan hệ giữa các dữ kiện.

ví dụ.

friends(X,Y) :- likes(X,Y),likes(Y,X).	/* X và Y là bạn bè nếu họ thích lẫn nhau */
hates(X,Y) :- not(likes(X,Y)).	/* X ghét Y nếu X không thích Y. */

Logic atom:<nguyên tử logic>: biểu diễn quan hệ giữa các hạng (term) -> Hạng và quan hệ của hạng tạo thành một mệnh đề.

Term: đối tượng dữ liệu của prolog. Hạng sơ cấp: hằng(constant), biến (variable).

Hạng phức hợp: biểu diễn các đối tượng thực tập của bài toán. Hạng phức hợp là một hàm tử functor chứa các đối số argument.

Functor : Tên_hàm_tử (Đối_1, Đối_2,... ,Đối_n).

Query: dùng để tra cứu một điều gì đó. Ví dụ Trinh thích Trung.

?- likes(Trinh,Trung).

Sau khi hiển thị câu trả lời đầu tiên .Prolog sẽ lần lượt tìm kiếm dữ liệu thỏa mãn và lần lượt hiện thị kết quả nếu chừng nào người còn yêu cầu cho đến khi không còn kết quả cho câu query đó nữa.

Ví dụ.

?- weather(City,summer, hot).

X= hcm;

X=dn;

ở đây kết quả đầu tiên, để tiếp tục nhận kết quả khác, người dùng tiếp tục yêu cầu bằng dấu hai chấm (;). Nhấn enter hoặc dấu chấm để kết thúc luồng trả lời.

Trường hợp nếu người dùng có thể dùng symbol để kết hợp các câu hỏi lại với nhau.

Ví dụ câu hỏi x và y có cùng cha không?

?- father(X,x), father(X,y)

Prolog sẽ tìm một người X thỏa mãn vừa là cha của x và y. Không có thì sẽ trả lời no,

Chú thích: /*chú thích*/ để chú thích nhiều dòng hoặc % để chú thích theo dòng.

Các kiểu dữ liệu trong prolog.

Kiểu dữ liệu	Ví dụ
Kiểu số nguyên và số thực	1 1.23 -0.45 -1.0
Kiểu logic	True false
Kiểu chuỗi	“Nguyễn Phước Quang” “” // chuỗi rỗng “/” // chuỗi chỉ có nháy kép

Đệ qui trong prolog.

Để xác định tổ tiên trong cây phả hệ. Ta cần xác định nhiều luật theo nhiều bậc của cây phả hệ.

Ví dụ: `ancestor(X,Y):parents(X,Z),parents(Z,Y).`

1.3 Các phép toán trong prolog.

Cộng +.

Trừ -.

Nhân *.

Chia cho số thực /.

Chia lấy số nguyên //.

Mod chia lấy phần dư.

Biểu thức số học được xây dựng bằng vị từ `is`. Đối số bên trái là đối số sơ cấp còn đối số bên phải là đối số biểu thức toán học.

Ví dụ: `?- X = 1+2.`

`X = 1+2.`

`?- X is 7*2.`

`X = 14.`

Prolog có hỗ trợ các hàm tính sẵn như `cos`, `sin`, `tan`, `sqrt`...

Ví dụ: `-? X is cos(90).`

`X = -0.4480736161291701.`

Ký hiệu prolog	Ý nghĩa
<code>></code>	Lớn hơn
<code>>=</code>	Lớn hơn hoặc bằng
<code><</code>	Nhỏ hơn
<code>=<</code>	Nhỏ hơn hoặc bằng
<code>=\=</code>	Khác
<code>=:=</code>	Bằng nhau

sánh

between(a,b,Z)	$a < Z < b$
Plus(Int1,Int2, Int3)	Thành công khi $\text{Int3} = \text{Int2} + \text{Int1}$

Các phép so
trong prolog.

1.4 Cấu trúc dữ liệu trong prolog.

Danh sách là một cấu trúc dữ liệu được tạo dựng sẵn trong SWI Prolog và cũng đã có sẵn các phép toán để lấy phần tử đầu và phần đuôi danh sách. Danh sách là nhóm bất kỳ các hạng thức với nhau bằng dấu “[“ và “]” và phân cách bởi dấu “,”. Ví dụ [a,b,c,d] là danh sách gồm 4 phần tử. Thao tác cơ bản để thao tác với danh sách là tách phần tử đầu của danh sách. Ví dụ:

1 ? – [X|Y]=[a,b,c,d].

X=a, Y=[b,c,d]

2 ? – [X,Y|Z]=[a,b,c,d].

X=a, Y=b, Z=[c,d]

3 ? – [X,[Y|Z]]= [a,b,c,d].

X=a, Y=b, Z=[c,d].

Ngoài thao tác cơ bản ở trên, SWI cũng đã xây dựng một số thao tác khác.

Ví dụ: 4 ? – member(b,[a,b,c,d]). % b có phải là phần tử của danh sách [a,b,c,d] không?

True

5 ? – append([a,b,c],[d,e,f],X). % nối hai danh sách X = [a, b, c, d, e, f]

2. Tìm hiểu môi trường lập trình prolog.

2.1 Giới thiệu SWI-Prolog.

SWI-Prolog là một công cụ lập trình Prolog được sử dụng rất phổ biến, có các phiên bản chạy trên các hệ điều hành Windows, MacOS, và Linux.

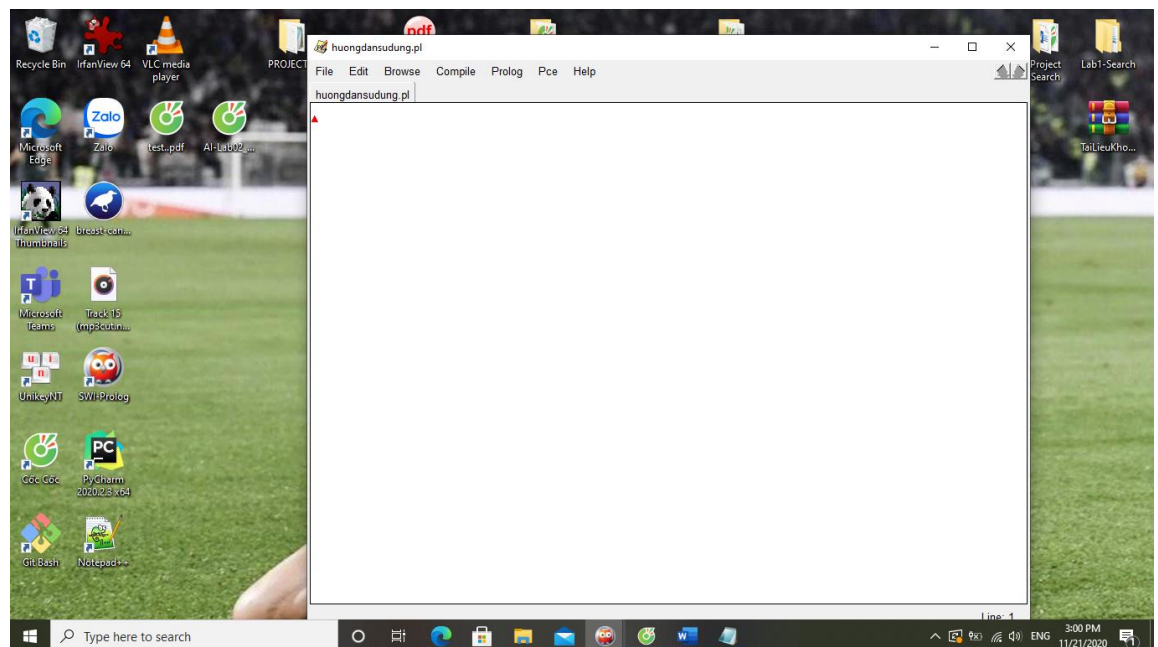
Bản quyền sử dụng phần mềm SWI-Prolog là miễn phí cho mục đích học tập và nghiên cứu.

SWI-Prolog có thể tải về từ địa chỉ: <http://www.swi-prolog.org/>.

2.2 Sử dụng SWI – Prolog.

Mỗi chương trình Prolog là một cơ sở dữ liệu gồm các mệnh đề. Các mệnh đề trong chương trình có thể được sắp xếp theo bất kỳ trật tự nào. Các mệnh đề của một vị từ được sử dụng theo đúng trật tự của chúng trong chương trình.

Để sử dụng chương trình soạn thảo để tạo ra cơ sở tri thức. Ghi lại chương trình trong một tập tin định dạng văn bản sử dụng đuôi của tập tin là .pl. Chúng ta mở SWI Prolog rồi chọn hộp thoại File tiếp theo Edit.



Ví dụ : Trung thích Nhi.

Nhi thích Trung.

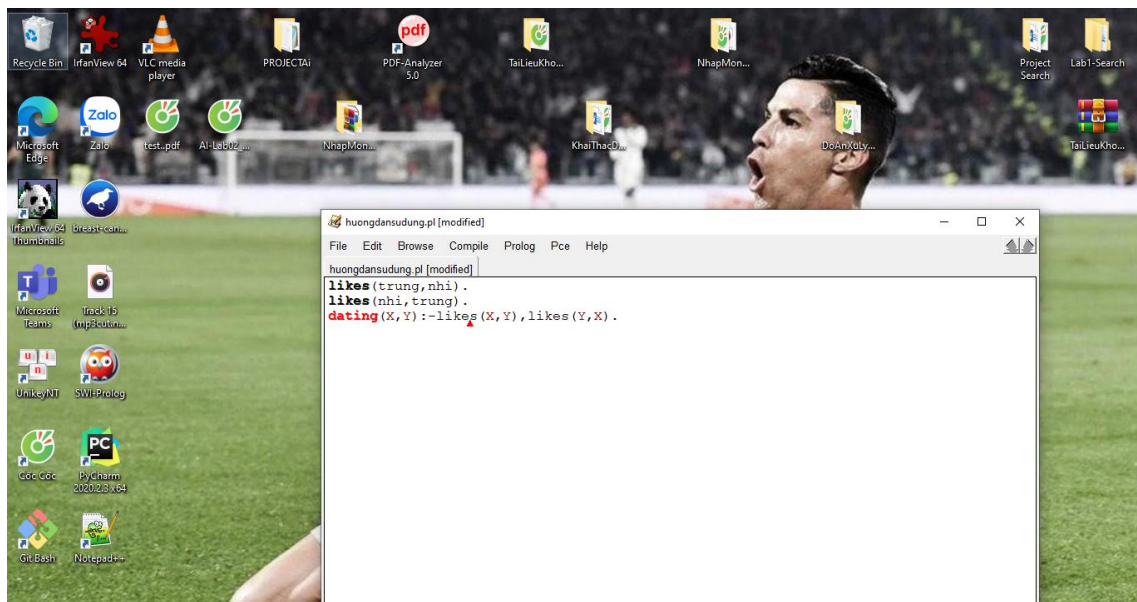
Hai người hẹn hò nếu hai người đó thích lẫn nhau.

Tạo các vị từ và mệnh đề .

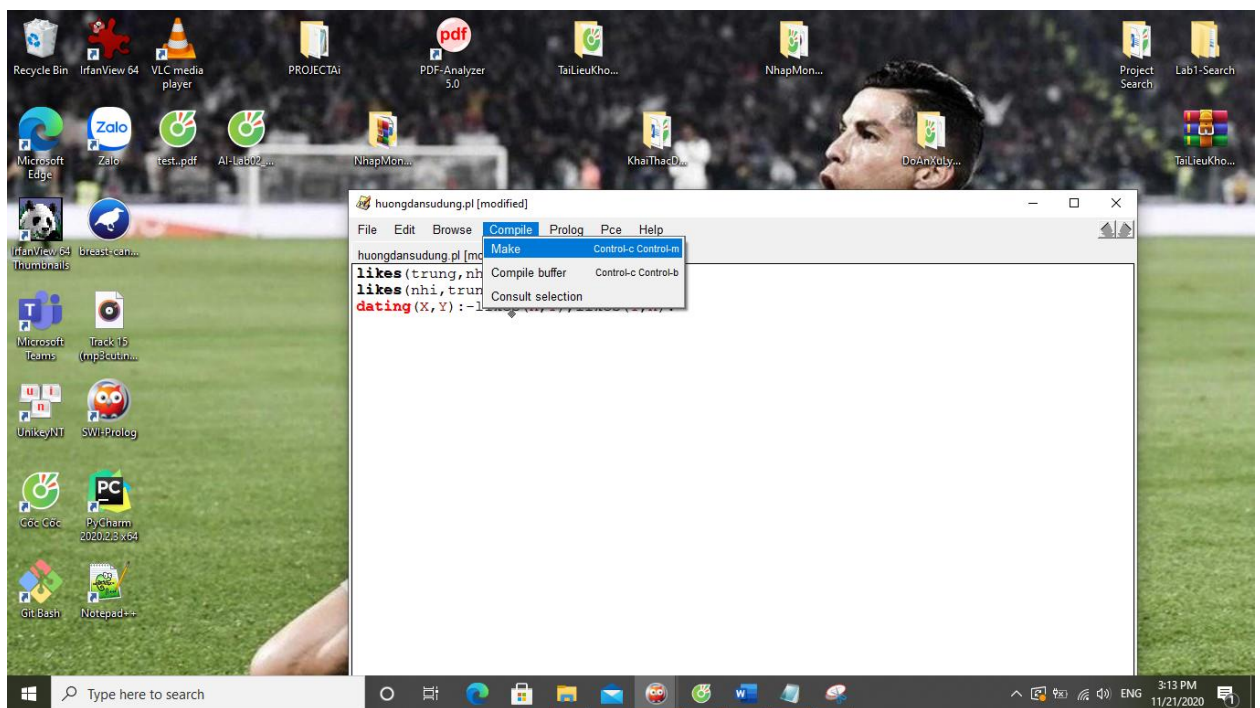
likes(trung,nhi).

likes(nhi,trung).

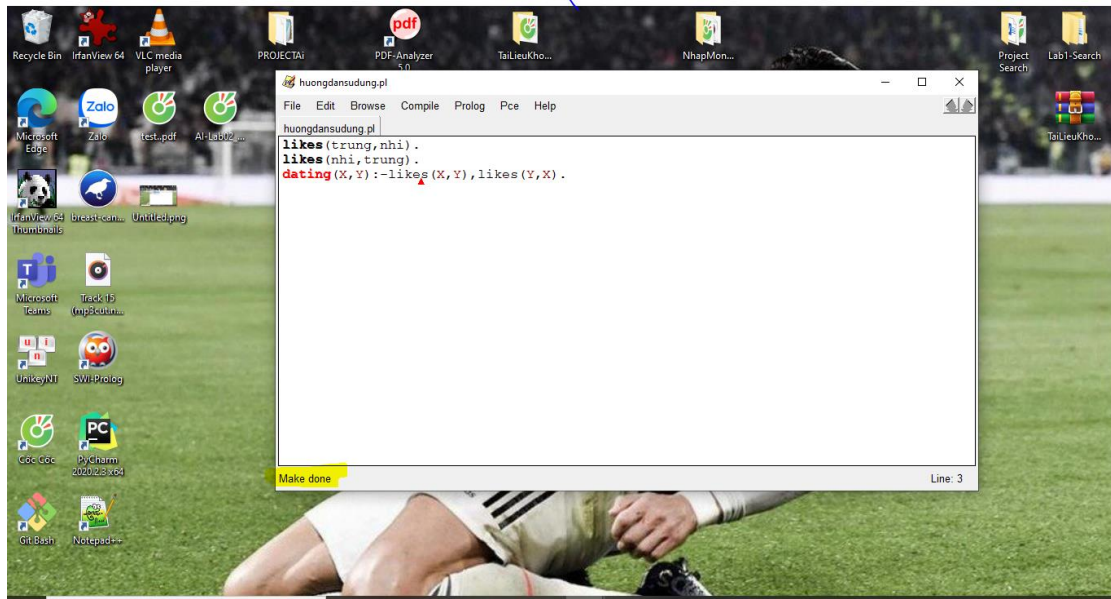
dating(X,Y):- likes(X,Y) , likes(Y,X).



Để chạy đường chương trình trước tiên chúng ta cần compile chương trình. Chúng ta chọn hội thoại compile rồi chọn Make.

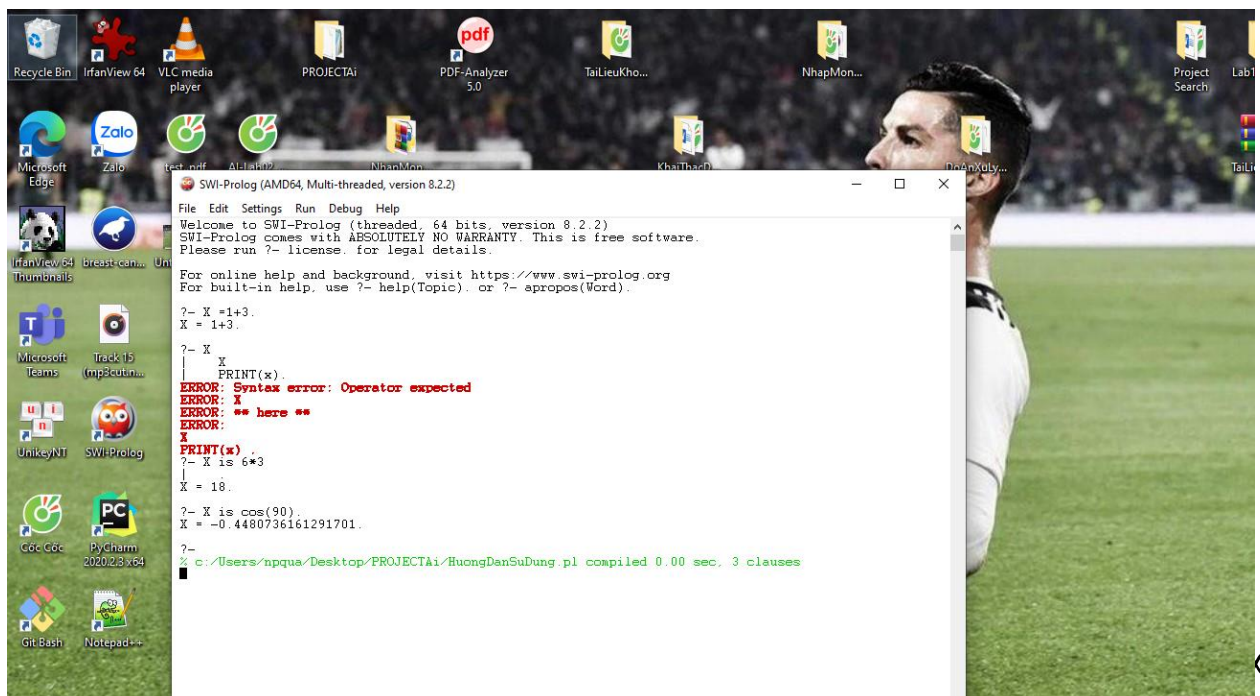


Chương trình không có loại góc trái bên dưới sẽ báo make done.



Nếu chương trình có lỗi bạn cũng có thể compile được. Vấn đề này mình sẽ nói trong phần sau.

Tiếp tục để thực hiện các câu truy vấn ta chọn hội thoại File rồi chọn consult .



Ví dụ :- likes(trung,nhi).

true.

?- likes(quang,linh).

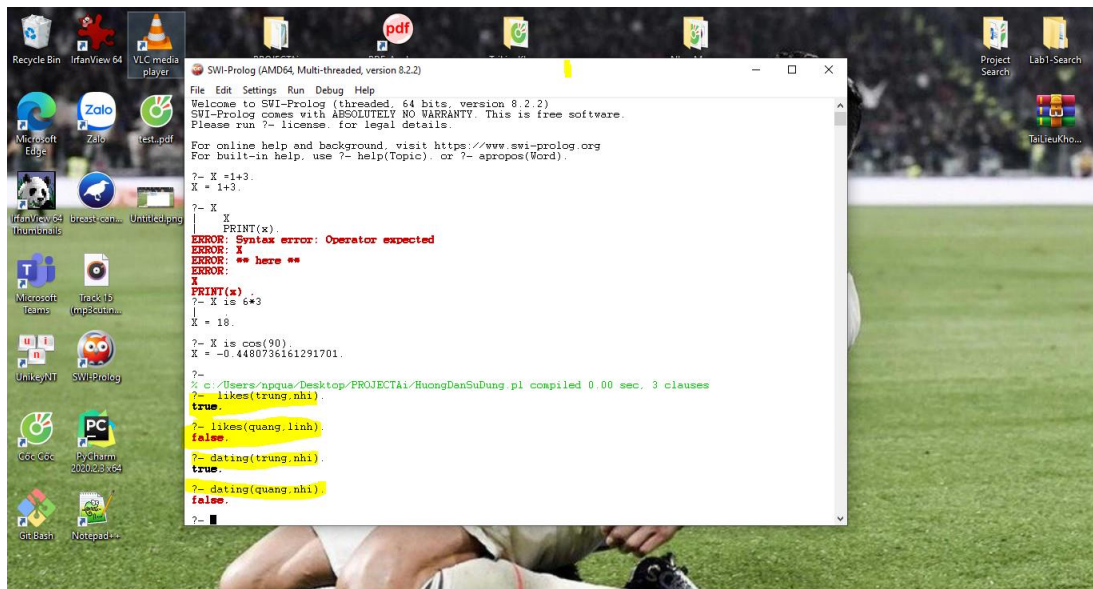
false.

?- dating(trung,nhi).

true.

?- dating(quang,nhi).

false



2.3 Cú pháp prolog trong SWI – Prolog.

Một cấu trúc gồm tên và không , một hoặc nhiều tham số.Nếu không có tham số thì bỏ đi dấu ngoặc.

Một cấu trúc chính là mệnh đề cơ sở . Một mệnh đề cơ sở biểu diễn một sự kiện.

Ví dụ.

trung.

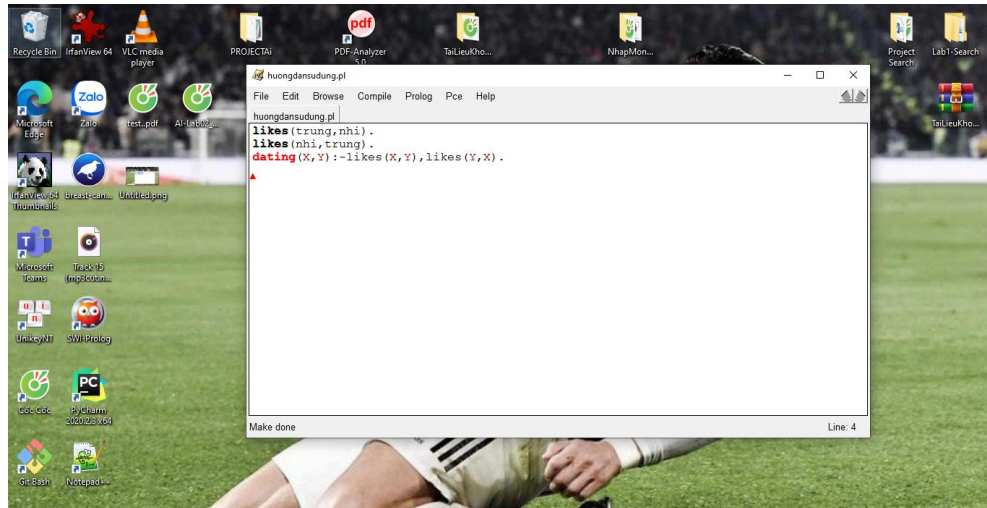
nhi.

likes(trung, nhi).

Một luật được biểu diễn bao gồm:

- Một mệnh đề kết luận của luật.
- Ký hiệu : :-

- Một danh sách các cấu trúc mệnh đề giả thiết của luật được kết nối với bảng toán tử logic .



Biến và hằng.

- Các biến bắt đầu bằng chữ cái in hoa hoặc ký tự đặt biệt.
Ví dụ: X , Name, _result.
- Các hằng không bắt đầu bằng các chữ in hoa hoặc các ký tự đặt biệt.
Ví dụ: x, name.

Kiểu số.

- Các số nguyên được sử dụng nhiều trong Prolog.
- Cú pháp của nó rất đơn giản: 365.-100.0, 3.36.

Các chú ý khi sử dụng SWI -Prolog.

- **Phân biệt chữ hoa và chữ thường là rất quan trọng.**
- **Không được có khoảng trắng giữa tên và danh sách của một trúc.**
- **Kết thúc một mệnh đề phải có dấu chấm.**

3. Xây dựng cơ sở tri thức cây phả hệ cho gia đình Hoàng gia Anh.

3.1 Định nghĩa và suy diễn vị từ.

Nhóm vị từ định nghĩa sẵn gồm: parent(Parent,Child), male(Person), female(Person), married(Person,Person), divorced(Person,Person).

Với nhóm định vị từ này, lưu ý mối quan hệ tình trạng hôn nhân cần định nghĩa cả :
married(X,Y) và divorced(X,Y).

Nhóm vị từ suy diễn:

husband(Person,Wife): Person là nam và đang kết hôn với đối tượng Wife.

wife(Person,Husband): Person là nữ và đang kết hôn với đối tượng Husband.
 father(Parent,Child): Parent là nam và cha của đối tượng Child.
 mother(Parent,Child): Parent là nữ và mẹ của đối tượng Child.
 Child(Child,Parent): Child có phụ huynh của đối tượng Parent.
 son(Child,Parent): Child là nam và con của đối tượng Parent.
 daughter(Child,Parent): Child là nữ và con của đối tượng Parent.
 grandparent(GP,GC): GP là phụ huynh của phụ huynh của đối tượng GC.
 grandmother(GM,GC): GM là nữ và ông bà của đối tượng GC.
 grandfather(GF,GC): GF là nam và ông bà của đối tượng GC.
 grandchild(GC,GP): GC có ông bà của đối tượng GP.
 grandson(GS,GP): GS là nam và là cháu của đối tượng GP.
 granddaughter(GD,GP): GD là nữ và là cháu của đối tượng GP.
 sibling(Person1,Person2): Person1 khác Person2, cha của Person1 và Person2 giống nhau của Person1 và Person2 cũng giống nhau.
 borther(Person,Sibling): Person là nam và anh/ chị/em ruột của Sibling.
 sister(Person,NieceNephew): Person là nữ, là chị/em ruột với Sibling.
 aunt(Person,NieceNephew): Person là nữ, là chị/em của ba/mẹ hoặc là vợ của chú NieceNephew.
 uncle(Person, NieceNephew): Person là nam và có dì/chú là NieceNephew.
 niece(Person,AuntUncle):Person là nữ và có dì/chú là AuntUncle.
 nephew(Person,AuntUncle): Person là nam và có dì/chú là AuntUncle.

3.2 Bộ câu hỏi.

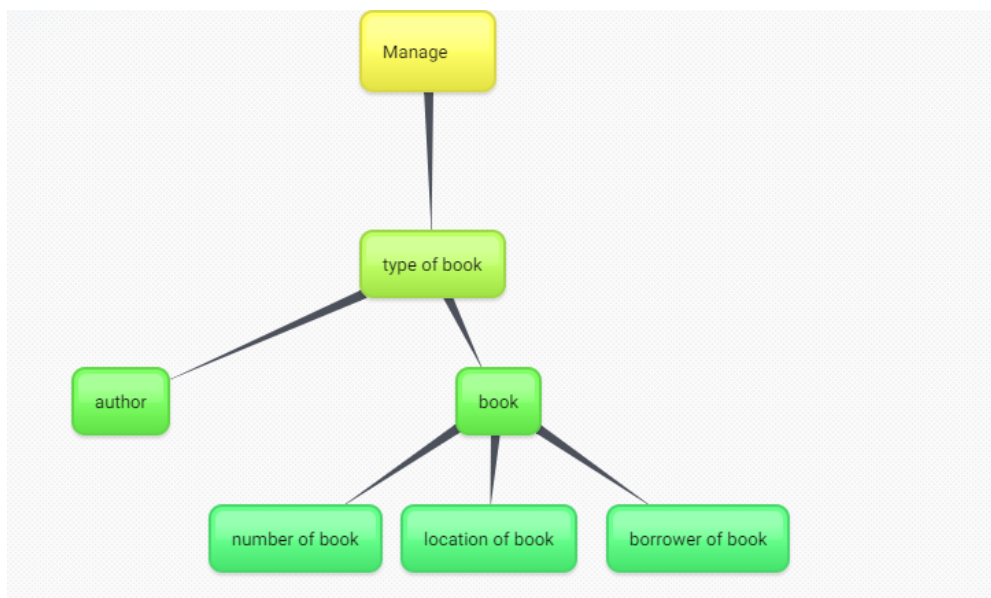
- 1 Ai là mẹ của Prince Andrew ?
- 2.Nữ hoàng Elizabeth có phải là vợ của Mia Grace Tindall không ?.
- 3 Con của Peter Philips là ai ?
- 5 Ai là cha của Prince Will ?
- 7 Sarah Ferguson có còn là vợ của Prince Andrew không ?
- 8 Ai là anh em của pWilliam ?
- 9 Ai là cháu gái của Princess Diana ?
- 10 Ai là cha mẹ của Mia Grace Tindall ?
- 11 Ai là cha mẹ của Peter Phillips ?
- 12 Autumn Kelly có phải là vợ của Mike Tindall không ?
- 13 Princess Beatrice có phải là con của Prince Phillip không ?
- 14 Ai là cháu gái của Princess Anne ?
- 15 Ai là cháu trai của Princess Diana ?
- 16 Ai là chồng của Pricess Diana ?
- 17 Ai là cha của Prince William ?
- 18 Ai là cô/dì của Price Harry ?
- 19 Mia Grace Tindall có phải là con của Mike Tidall không ?
- 20 Ai là con của Kate Middleton ?

4. Xây dựng cơ sở tri thức quản lý thư viện với công cụ Prolog.

4.1 Định nghĩa và nhóm vị từ.

Vị từ	Định nghĩa
manage(X)	Người quản lý thư viện
type(X)	Các loại sách trong thư viện
manage_author(X,Y)	Y là tác giả thuộc thể loại X
manage_book(X,Y)	Y là sách thuộc thể loại X
book(X,Y)	X là tác giả của cuốn sách Y
number_book(X,Y)	Y là số lượng sách của X
borrower(X,Y)	X là người mượn cuốn sách Y
location(X)	X là khu vực để sách
location_book(X,Y)	Y là cuốn sách được đặt ở khu vực X
find_author(X,Z)	X: là tên tác giả , Z là tên cuốn sách.
lovefield(X,Y)	X là tên người mượn, và Y là thể loại sách. Lovefield tìm kiếm thể loại sách người đó hay đọc.
find_loaction(X,Y)	X: là tác giả , Y là khu vực để sách.
numberbook(X,Y,N)	X là thể loại sách. Y là tên cuốn sách. N là số lượng của cuốn sách.
checking(X,Z,R)	X là tên tác giả. Z,R là thể loại sách.

4.2 Cở sở dữ liệu và sơ đồ.



Vị từ	Dữ liệu
Người quản lý	Nguyenphuocquang
Thể loại sách	self_development_book history_book geography_book science_book
Tác giả	nguyenvanA nguyenvanB nguyenvanC) nguyenvanD nguyenvanF nguyenvanG nguyenvanH nguyenvanL
Sách thuộc thể loại self_development_book	book_sdb_a book_sdb_b book_sdb_c book_sdb_d
Sách thuộc thể loại history_book	book_hb_a book_hb_b book_hb_c
Sách thuộc thể loại geography_book	book_gb_a book_gb_b

	book_gb_c
Sách thuộc thể loại science_book	book_sb_a book_sb_b book_sb_c
Số lượng của mỗi cuốn sách	10

4.3 Bộ câu hỏi.

- 1 Ai là người quản lý thư viện.?
- 2 Tất cả các tác giả trong thư viện.?
- 3 levanL có phải là tác giả không ?
- 4 Tất cả cuốn sách trong lĩnh vực self_development_book.
- 5 Sách của tác giả nguyenvanA là những cuốn sách nào ?
- 6 Thông tin tất cả các người mượn sách trong thư viện ?
- 7 Tất cả cuốn sách ở location_a là những cuốn sách nào ?
- 8 nguyenvanD có viết sách về lĩnh vực history_book không ?
- 9 Tất cả tác giả viết về thể loại self_development_book ?
- 10 Những lĩnh vực của tác giả nguyenvanA tham gia viết ?
- 11 Chi tiết tên sách , số lượng từng cuốn sách về lĩnh vực history_book ?
- 12 levanB có thích đọc sách về history_book hay không ?
- 13 lethiA sẽ mượn những cuốn sách về những lĩnh vực nào ?
- 14 cuốn sách book_gb_b thuộc thể loại nào ?
- 15 Cuốn sách book_sb_a được lưu trữ ở đâu ?
- 16 Sách của tác giả nguyenvanB để ở khu vực nào ?
- 17 Thông tin người mượn cuốn sách book_hb_b ?
- 18 Thông tin tất cả những người mượn sách ?
- 19 nguyenvanD có phải là tác giả không?
- 20 Tất cả cuốn sách về science_book ?

5. Cài đặt hệ thống suy diễn logic bằng ngôn ngữ lập trình.

- Ngôn ngữ lập trình : Python
- Phương pháp suy diễn: Suy diễn tiến.
- Đầu vào, đầu ra và cú pháp tương tự như Prolog
- Phương pháp và các điểm chưa tối ưu.
 - Phương pháp:
 - Phân tích nhỏ các câu hỏi (đích) dựa theo các luật cho vào, cho đến khi không thể phân tích thêm.
 - Các câu hỏi bây giờ được so sánh với các mệnh đề cho trước, nếu tồn tại đúng thì câu hỏi ban đầu là đúng và ngược lại.
VD: Clause : $f(a)$, $f(b)$.
Rule: $g(x) :- f(x)$.
Goal: $g(a)$
Phân tích $g(a)$ thành $f(a)$ và $f(a)$ tồn tại đúng trong clause nên goal đúng.
 - Các điểm chưa tối ưu:
 - Không xử lý được với các luật mang tính đệ quy.
 - Không xử lý được với các câu hỏi chứa hai biến trở lên.
 - Không xử lý được với các luật mang tính phức tạp cao như:
 $A :- (B+C),(D+E)$. $A :- (B :- C)$
 - Xử lý khá ổn với các bài toán ở mức cơ bản như các trường hợp đề giao. (Cây phủ hệ)

```
(base) C:\Users\Admin>cd C:\Users\Admin\Downloads\1712704_lab02\script
(base) C:\Users\Admin\Downloads\1712704_lab02\script>python main.py
Enter file: bai02.pl
?- location_book(X,book_sb_a).
X = location_d ;
?-
```

- Hướng dẫn chạy
 - Cmd: python main.py
 - Sau đó nhập file prolog và chạy.